Visualization and Analysis of Transportation-Specific Datasets

Michael Hope, Vadim Sokolov, Hubert Ley

Transportation Research and Analysis Computing Center

Energy Systems Division

Argonne National Laboratory

Current Trends in Transportation Visualization

- APIs to Plot on Existing Mapping Websites
  - Google Maps API
  - Yahoo Maps API
  - Bing Maps Platform
  - OpenLayers
- Web-based interfaces to GIS visualization
  - ArcGIS Online
  - MangoMap
- Web-based interface to transportation models
  - Cube Cloud
  - Vista
  - Cattlab at University of Maryland

Advantages of Web-based Tools

- Lower Upgrade Cost
- Lower Software Maintenance Cost
- Access Data Anywhere
- Cross Platform
- Share Data
- Scalable Processing Power
- Scalable Storage Space

Underserved Capabilities of Existing Web-Based Tools

- 3D Rendering
  - Extra spatial analysis dimension
  - Presentation quality
- Time Set Visualization / Data Streaming
  - Extra temporal analysis dimension
  - Create videos
- Desktop Application Level Performance
  - Larger areas
  - Larger datasets
  - More visualization real-estate

Underserved Capabilities of Existing Web-Based Tools cont’d

- Open Source
  - Free, unlicensed
  - Anyone can modify
- Transportation Specific
  - Analytics useful to researchers
  - Compatible with popular transportation models
- Non-commercial cloud storage and processing
  - Security issues
  - Data privacy

TRACC – A National User Facility to Meet USDOT Advanced Computation Needs

- USDOT and USDOE transportation research programs, private industry, and state and regional transportation agencies are moving to simulation-based design and analysis for improvements in efficiency, economics, and safety
- Higher fidelity analysis in areas such as crashworthiness, aerodynamics, combustion, thermal management, weather modeling, and traffic simulation require access to state-of-the-art computational and visualization facilities
- Argonne expertise in high-performance computing and transportation system analysis provides the basis for a national HPC user facility and a focal point for computational research for transportation applications

Argonne
NATIONAL LABORATORY

Transportation Research and
Analysis Computing Center

Visualization and Analysis Tools at TRACC

TRANSPORTATION RESEARCH AND ANALYSIS COMPUTING CENTER
U.S. DEPARTMENT TRANSPORTATION
TRACC
LN

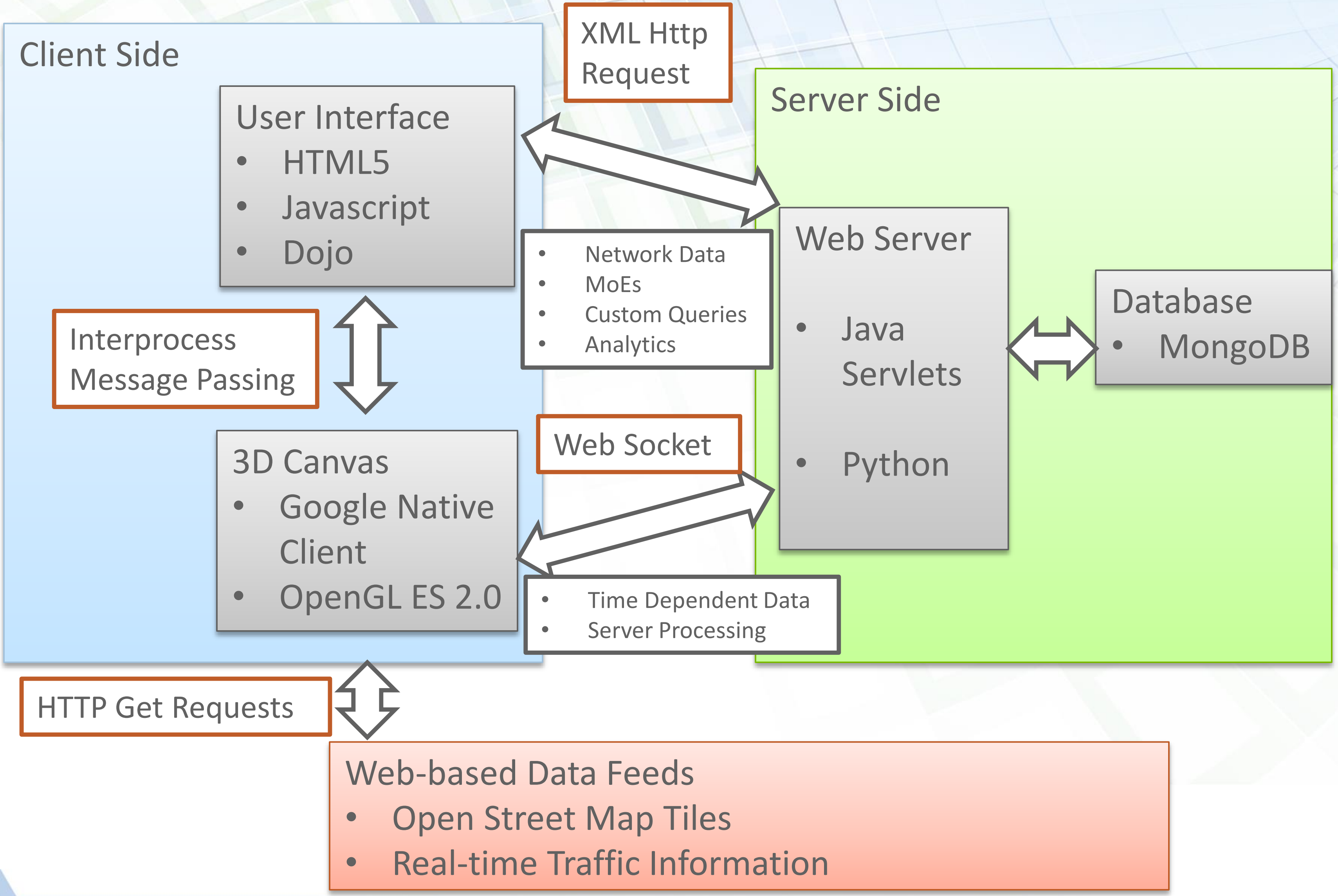
- Metroview
  - Web-Based Transportation Visualizer
- Antares
  - Simulation Graphics Library
- POLARIS Analyzer
  - Common Analytics on Transportation Datasets
- Network Editor
  - Cross-product Network Editing

METR VIEW

- Web-based Transportation Visualizer
  - 3D Visualization
  - Dynamic Time Series Visualization
  - Desktop-Level Performance
  - Open Source
  - Hosted at Argonne-TRACC

- Features:
  - Network Visualization
  - Link-based Measures of Effectiveness
  - Vehicle Trajectory Visualization
  - MongoDB Data Storage
  - Open Street Map Tiles

Metroview Architecture



Client Side

User Interface
- HTML5
- Javascript
- Dojo

Interprocess Message Passing

3D Canvas
- Google Native Client
- OpenGL ES 2.0

XML Http Request

- Network Data
- MoEs
- Custom Queries
- Analytics

Web Socket

- Time Dependent Data
- Server Processing

Server Side

Web Server
- Java Servlets
- Python

Database
- MongoDB

HTTP Get Requests

Web-based Data Feeds
- Open Street Map Tiles
- Real-time Traffic Information

Metroview - MongoDB

- Non-SQL Database
  - Redesigned Query Language
- Document-Oriented Storage
  - JSON Format
  - Extremely Flexible
- Higher Performance, Scalable
  - Larger Queries
  - More Users
- Multiple Drivers
  - Python, Java, C++

Json Example

```
Links = [

{ “ LinkID“ : “1" ,
                “Delay“ : [ “ AM“ : “4" , “PM“ : “5“] ,
                “Shape“ : “GEODATA“ },

{ “ LinkID“ : “2" ,
                “Delay“ : [ “ MID “ : “4“] },

{ “ LinkID“ : “3" ,
                “Delay“ : [ “PM“ : “5“] ,
                “Shape“ : “GEODATA“ }

];
```

Metroview – Google Native Client

- Novel Web Framework by Google
  - In Active Development
- Run C++ Programs Through a Browser
- OpenGL ES 2.0 Graphics
- Near Desktop Level Performance
- Robust API
  - Multithreading
  - HTTP Requests
  - File IO

Metroview – Internal Task Scheduler

- Advanced Task Management – automated multithreading

- Submit Work to 2 Types of Queues:
  - Traditional “immediate” work queue
  - Time-dependent queue with “execute after” and “execute by” time

- Time Independent Queue
  - Performs work in FIFO order

- Time Dependent Queue
  - Used in conjunction with visualizer time stepping
  - Visualizer states what time it would like to advance to and authorizes it
  - Flexible scheduling allows significantly better load balancing
  - Automatic merging of requests when scheduling permits

- Time Dependent Threads can Migrate to Time Independent Work

Metroview – Map Tile Loading

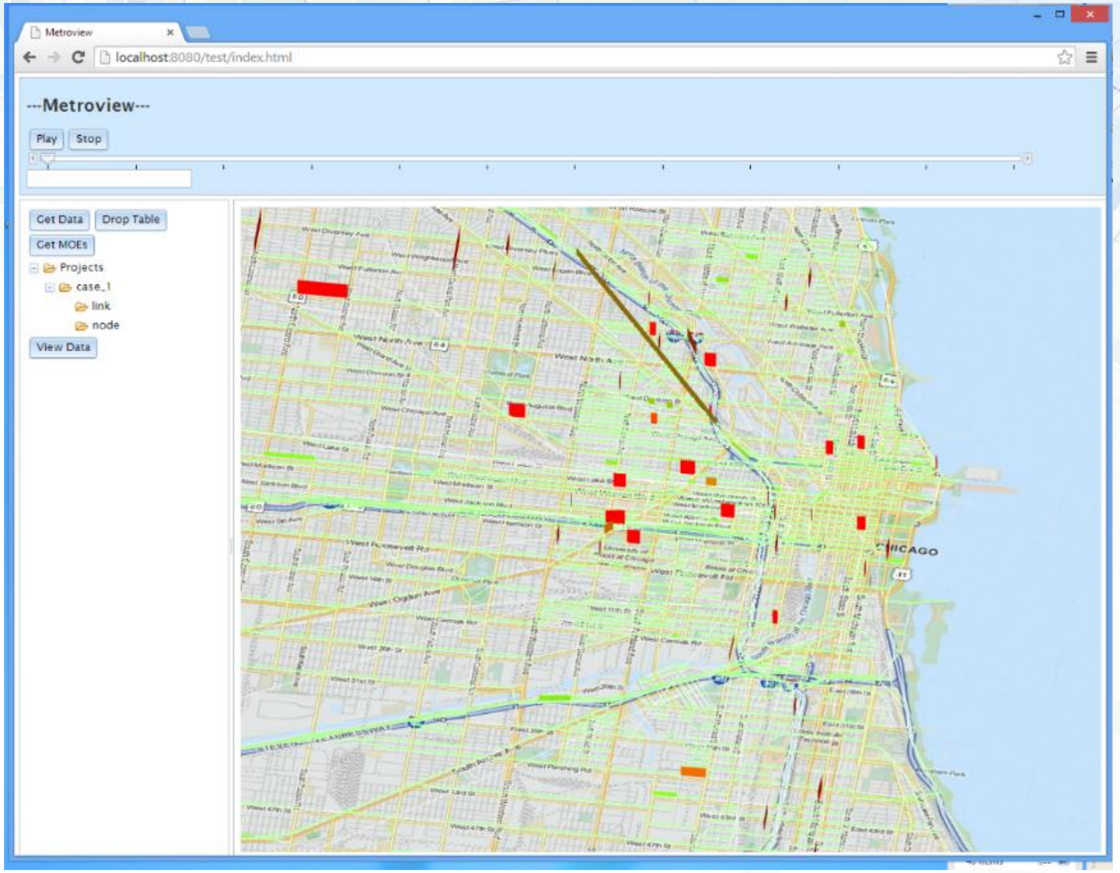
- Currently utilizing MapQuest-OSM Tiles

- Soon to be compatible with MapQuest Open Aerial Tiles

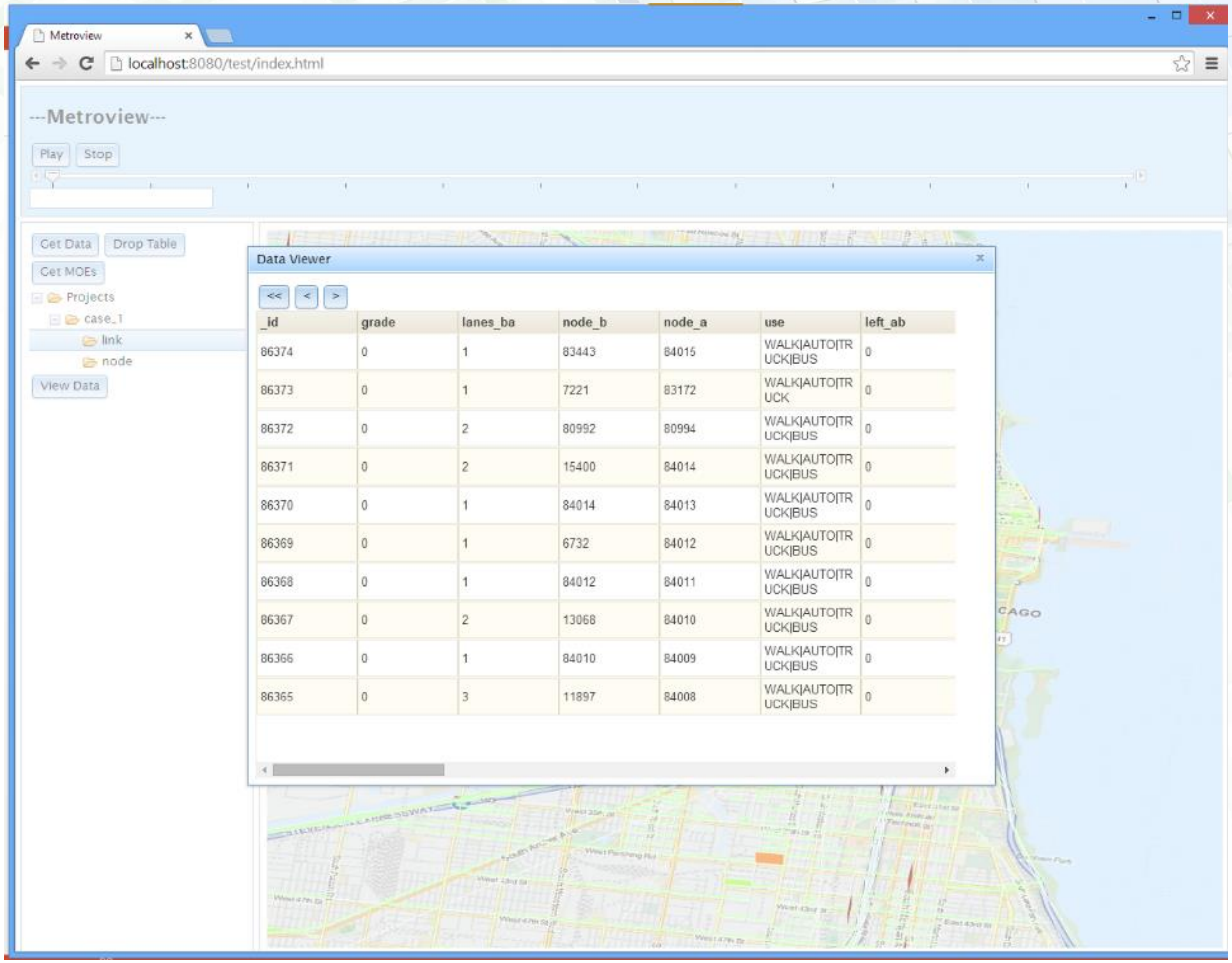
- Downloaded as you navigate

- Compatible with Slippy Map Tilenames
  - MapQuest
  - MapQuest Open Aerial
  - OSM standard
  - OpenCycleMap
  - Migurski's Terrain

- Underlying EPSG:3857 projection
  - Seamless navigation across UTM zones
  - Properly aligned tiles



Metroview - Miscellaneous



_id	grade	lanes_ba	node_b	node_a	use	left_ab
86374	0	1	83443	84015	WALK|AUTO|TRUCK|BUS	0
86373	0	1	7221	83172	WALK|AUTO|TRUCK	0
86372	0	2	80992	80994	WALK|AUTO|TRUCK|BUS	0
86371	0	2	15400	84014	WALK|AUTO|TRUCK|BUS	0
86370	0	1	84014	84013	WALK|AUTO|TRUCK|BUS	0
86369	0	1	6732	84012	WALK|AUTO|TRUCK|BUS	0
86368	0	1	84012	84011	WALK|AUTO|TRUCK|BUS	0
86367	0	2	13068	84010	WALK|AUTO|TRUCK|BUS	0
86366	0	1	84010	84009	WALK|AUTO|TRUCK|BUS	0
86365	0	3	11897	84008	WALK|AUTO|TRUCK|BUS	0

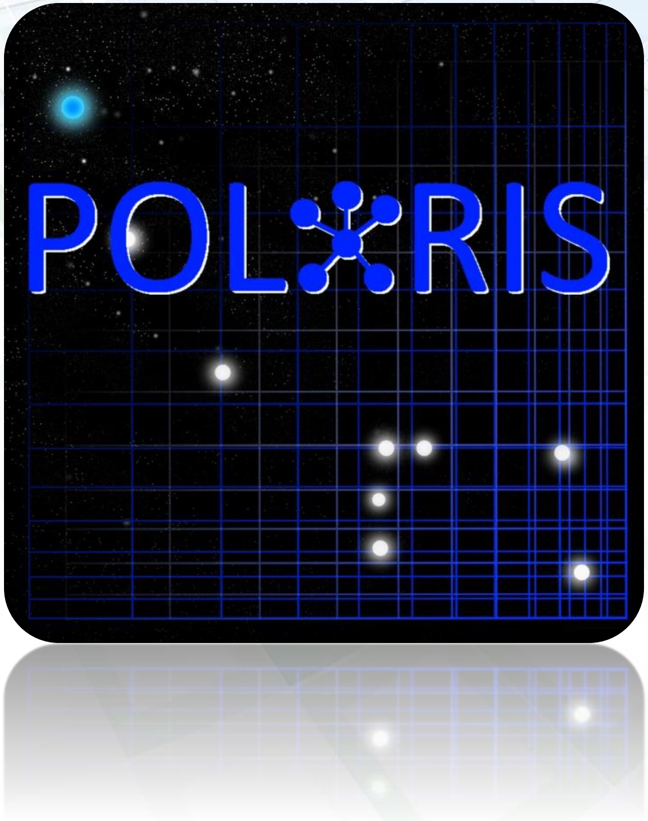
- Tabularized Data Viewer
- Websocket Server Connectivity
  - Extremely low latency
  - Faster downloads
  - Very reliable connection
- DOJO
  - Popular Javascript UI Library
  - Many Standard Widgets
  - High Performance Widgets
- HTML5
  - Local Storage
  - Video / Audio

Metroview – Future Directions

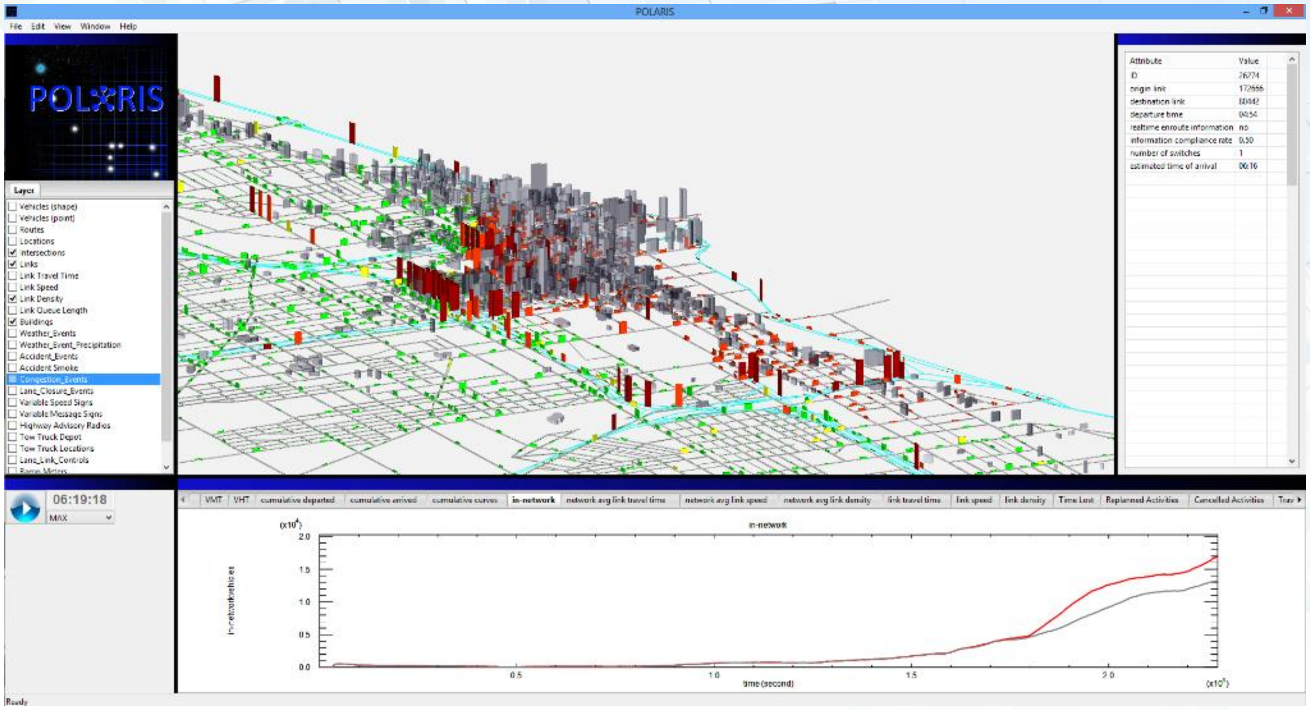
- Side-by-side Comparisons of Multiple Network Elements
- Server-side Analytics
  - Statistics
  - Regression Analysis
  - 2D Plots
- Run Custom Python User Scripts
- Data Conversions from Popular Tools
  - Vissim
  - Transmodeler
  - DTALite
  - DynusT
  - Etc…
- Real World Data Visualization
  - GPS Traces
  - Open Traffic Information

What is POLARIS?

- Open Source Middleware for Developing Agent-based Models
  - Data Interchange
  - Visualization
  - Case Study Generation and Analysis
  - Discrete Event Simulation
  - Interprocess Communication
- A Repository of Transportation Libraries
  - Common Algorithms
  - Extended by Researchers
  - Standardized Style and Structure
- Fully Developed Applications
  - Transportation Network Simulation
  - Integrated Activity Based Simulation



Antares



☐ Vehicles (shape)
☐ Vehicles (point)
☐ Routes
☐ Locations
☑ Intersections
☑ Links
☐ Link Travel Time
☐ Link Speed
☑ Link Density
☐ Link Queue Length
☑ Buildings
☐ Weather_Events
☐ Weather_Event_Precipitation
☐ Accident_Events
☐ Accident Smoke
☐ Congestion_Events
☐ Lane_Closure_Events
☐ Variable Speed Signs
☐ Variable Message Signs
☐ Highway Advisory Radios
☐ Tow Truck Depot
☐ Tow Truck Locations
☐ Lane_Link_Controls
☐ Ramp Meters

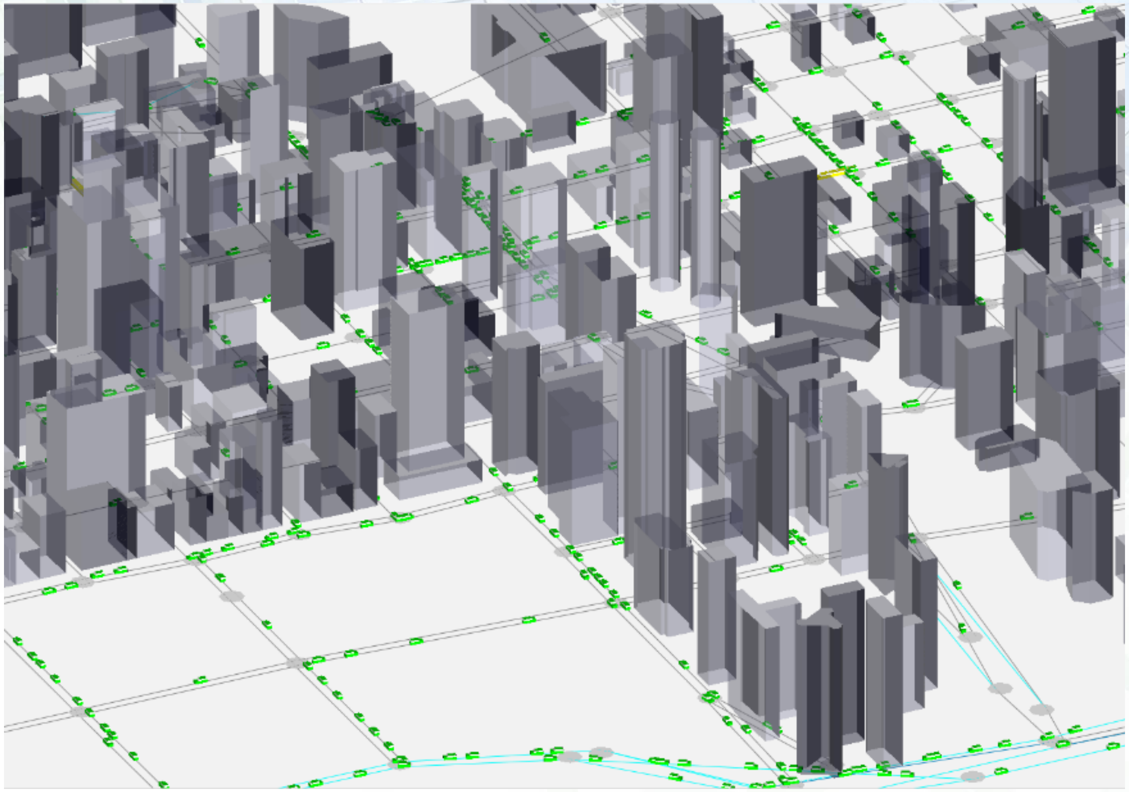
Attribute	Value
ID	76274
origin link	172666
destination link	80442
departure time	04:54
realtime enroute information	no
information compliance rate	0.30
number of switches	1
estimated time of arrival	06:16

- Graphics Library for POLARIS
  - Visualize in 3D or plot in 2D
  - See the real time state of your simulation
  - Inspect and change simulation elements on the fly
  - Create videos

- Technologies
  - WxWidgets
  - OpenGL
  - Plplot
  - FFMPEG

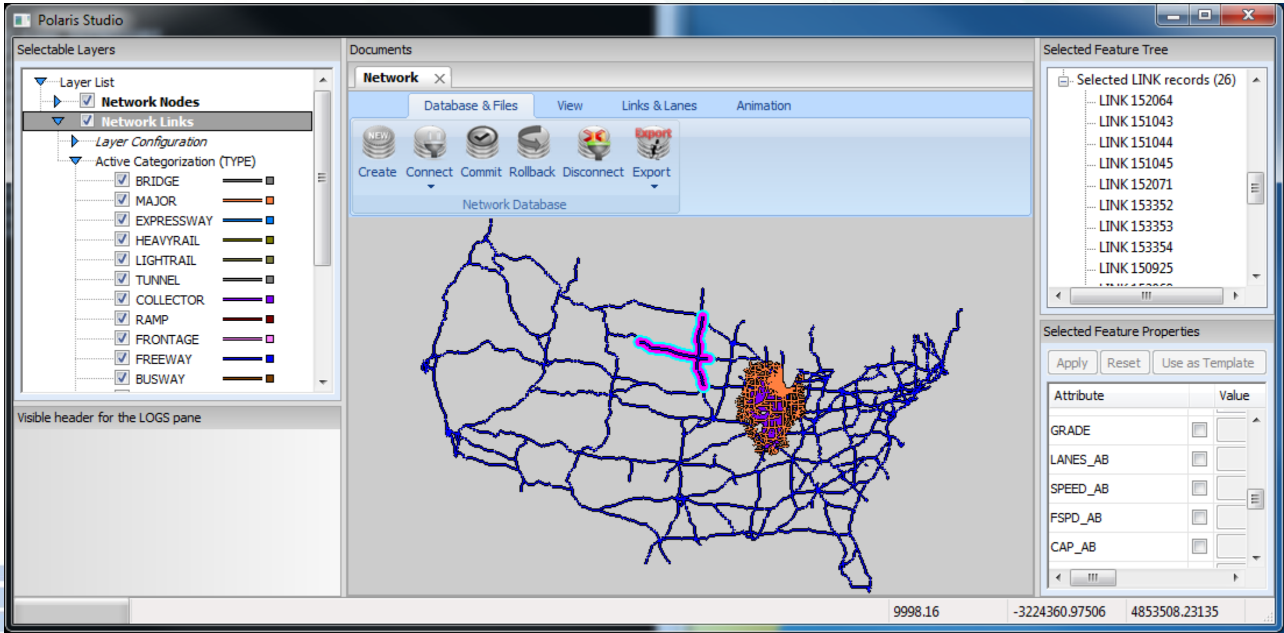
Antares - Developer Tools

- 3D Drawing Without Needing to Know OpenGL
- Organized Into Developer Allocated Layers
- Fully Integrated with Simulation and POLARIS Core Libraries
- Identification capabilities – automatically identify objects
- Set up events which trigger on clicking



POLARIS / TRANSIMS Network Editor

- The network is stored as a relational database using SQLITE3
- The network editor is just one of the applications using the database structure:
  - POLARIS tools access the network directly from the database
  - Quantum GIS is natively capable of working directly with the network database
  - Spatialite and Sqlite3 tools can be used to modify and edit the network as well
- The graphical user interface is both part of TRANSIMS Studio as well as Polaris
- Emphasis during development was to provide efficient editing tools that understand the network topology
- Background imagery is available at high resolution in a special format suitable for extremely fast editing



Selected LINK records (26)
LINK 152064
LINK 151043
LINK 151044
LINK 151045
LINK 152071
LINK 153352
LINK 153353
LINK 153354
LINK 150925

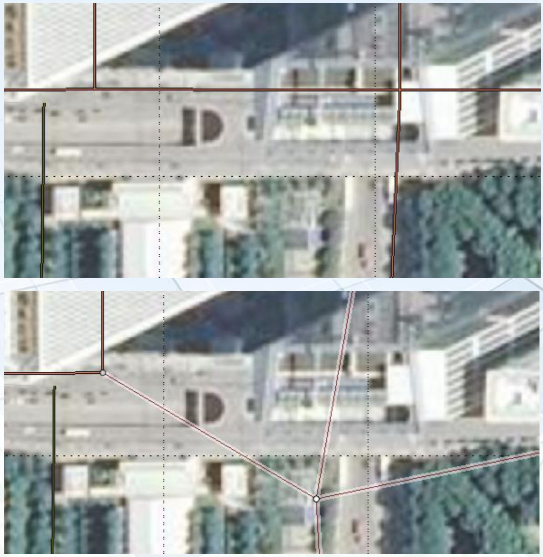
Attribute		Value
GRADE	□	
LANES_AB	□	
SPEED_AB	□	
FSPD_AB	□	
CAP_AB	□	

9998.16 -3224360.97506 4853508.23135

Orthorectified Imagery

- Orthorectified imagery from USGS is readily available at TRACC, with most imagery being at the most 2 years old (NAIP imagery)
  - NAIP dataset covers the entire US in full color (plus infrared)
  - Resolution is approximately 1 meter per pixel
- High resolution orthorectified imagery is available for certain urban areas at up to 025 meters per pixel
- The imagery datasets are processed on TRACC high performance computing systems and are provided in a highly efficient tile format to editing applications

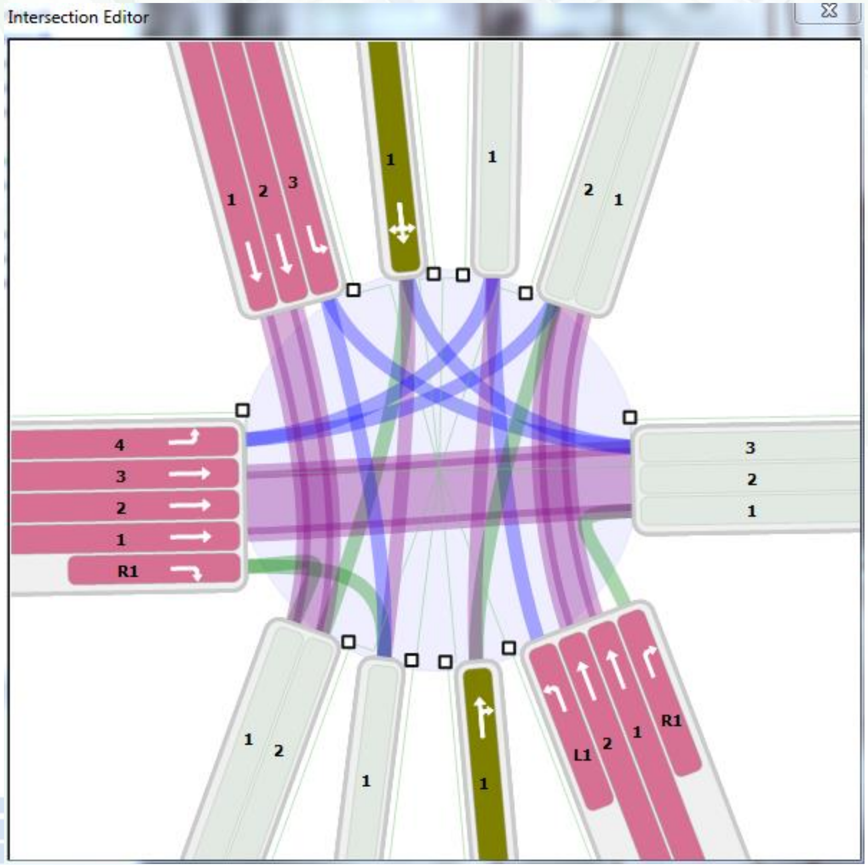
Network Topology



- The network editor understands the topology
  - Nodes, Links, Pockets, Connectivity, Signals, Signs, ...
- As shown on the right, moving a node will move all related objects as well
- The network database enforces relational integrity
  - This is an important feature, because it means that no client application, including the network editor, can actually break the relational integrity of the network
  - Changing any network features happens with a database transaction, meaning that the network can never be forced to be inconsistent
    - For example, a node cannot be deleted if a link still exists that uses this node
    - This is true throughout all relations – a total of about 60 relationships is enforced
- This is also an important feature when using third party tools such as Spatialite or Quantum GIS to work with the data
  - Relational integrity is enforced on the database level, preventing any client application from damaging relationships between network components

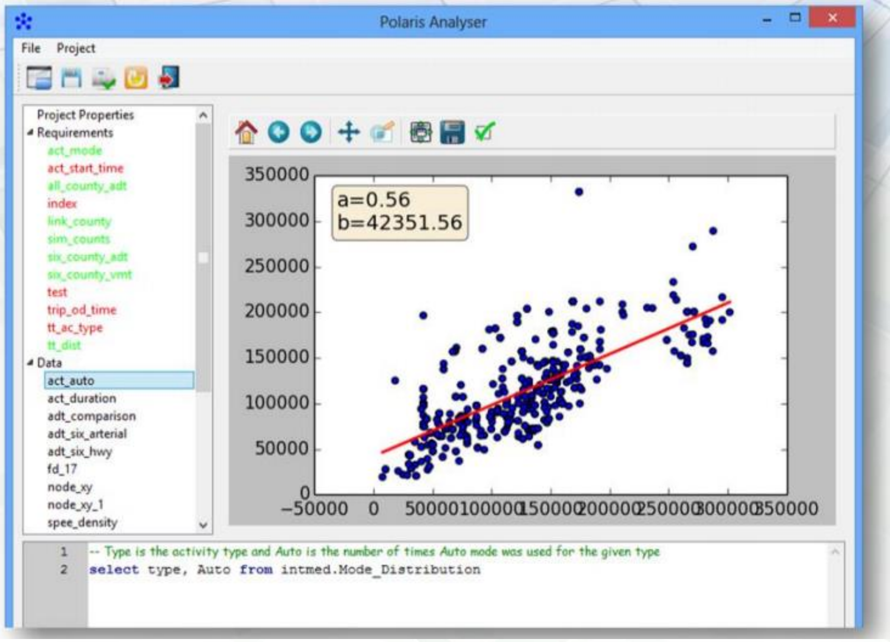
Intersection Editor

- Connectivity at intersections is highly configurable
- Rules can be defined based on facility type and similar criteria to establish a reasonable configuration that can then be fine-tuned for a specific intersection
- An example for Upper Wacker Drive and Monroe is shown on the right
  - Very complex intersection
  - Ramps from and to Lower Wacker Drive are fully represented
  - Connectivity is link to link, but allows for lane ranges inbound and outbound
- Groups of nodes can be selected in the general interface to apply automatic rules, e.g.
  - Reasonable choice of turn pockets based on lane permanent lane configurations
  - Thorough analysis of THRU connections, and proper choice of turn types



Data Analysis

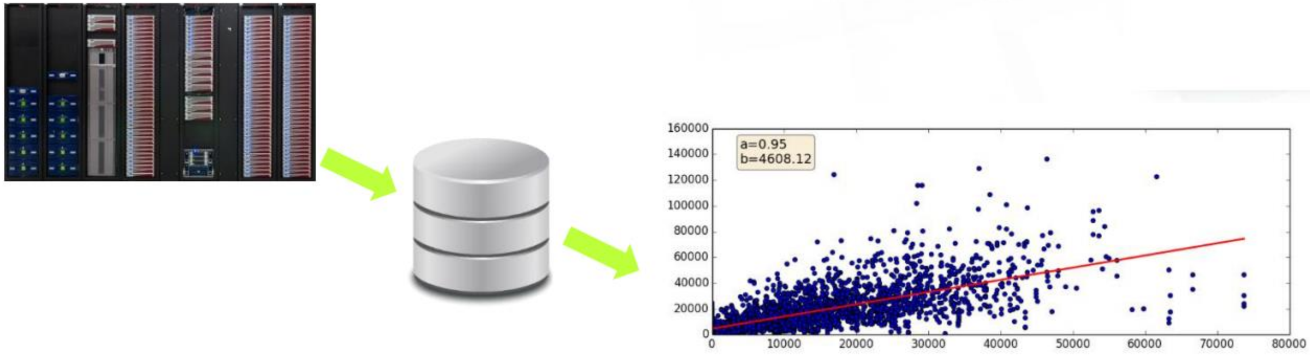
- Quickly, efficiently manage data
- GUI for analyzing simulation results
- Statistical analysis and data aggregation tasks
- Produce graphs and charts



```
-- Type is the activity type and Auto is the number of times Auto mode was used for the given type
select type, Auto from intmed.Mode_Distribution
```

Data Analysis

- Data is stored in a relational database with spatial capabilities
- User can define data processing steps in a form of
  - SQL Statement
  - Python script
- User can define type of analysis on each of the data items
  - Regression Analysis
  - Clustering Analysis
  - Exploratory Analysis (plots, charts)
- Procedures are stored in a form of a script and json file and can be applied to multiple data sets


a=0.95
b=4608.12
160000
140000
120000
100000
80000
60000
40000
20000
0
0
10000
20000
30000
40000
50000
60000
70000
80000